POCKET COMPUTER NEWSLETTER

© Copyright 1983 SCELBI Publications

CHANGING THE OUTPUT DISPLAY FORMAT

Some people using *SUPER SST* would like to have the results displayed to more than two decimal places. You can customize the program to your own tastes in this regards by proceeding as follows.

First, the USING statements in line 6130 and line 7000 must be altered to reflect the desired format. Say, for instance, that you wanted to be able to display six places after the decimal. Then the USING statement in these two lines would need to be redefined as:

USING "##,###,###.######"

Notice that you must not allow the total number of displayed digits to exceed the capability of the display. Thus, if you add four positions after the decimal point (to make a total of six), then you must take away four positions from the integer portion of the number. This means the order of magnitudes that can be displayed will be reduced by a corresponding number of digit positions.

Next, the rounding statements in line 4125 and 6170 must be altered to round to the desired number of decimal places. For instance, the statement at the end of line 4125 would need to be changed from its original form of:

A=INT(A*100+.5)/100

which rounds to two decimal places, to the form:

A=INT(A*1E6+.5)/1E6

which rounds to six decimal places. Don't forget to make the same change in the second statement of line 6170!

Listing *Changes to Display to Six or Four Decimal Places*

```
4125:Z$=X$+CHR$ (W):CLS :A=R(ASC (X$)-65,W-48):A=INT (A*1E6+.5)/1E6
4127:U=999999.9: IF ABS (A)>U LET A=-U
6130:USING "##,###,###.######":GLCURSOR (0,-200)
6170:A=R(I,L):A=INT (A*1E6+.5)/1E6:U=999999.9:IF ABS (A)>ULET A=-U
6175:LPRINT A
7000:ON ERROR GOTO 8050:USING "##,###,###.######":WAIT 0:CLS :PRINT " DEF LBL DIS I/O PRT CLR"
```

```
4125:Z$=X$+CHR$ (W):CLS :A=R(ASC (X$)-65,W-48):A=INT (A*1E4+.5)/1E4
4127:U=99999999.9 :IF ABS (A)>ULET A=-U
6130:USING "#####,###,###.####":GLCURSOR (0,-200)
6170:A=R(I,L):A=INT (A*1E4+.5)/1E4:U=99999999.9:IF ABS (A)>ULET A=-U
6175:LPRINT A
7000:ON ERROR GOTO 8050:USING "#####,###,###.####":WAIT 0: CLS :PRINT " DEF LBL DIS I/O PRT CLR"
```

Example *Printed Output to Six or Four Decimal Places*

```
COL A: START UP | COL B:    JAN 83
-999,999.900000 |      20.000000
-999,999.900000 |   1,900.000000
-999,999.900000 |     840.000000
-999,999.900000 |     180.000000
-999,999.900000 |   4,000.000000
-999,999.900000 |      70.000000
-999,999.900000 |     850.000000
-999,999.900000 |   5,940.000000
-999,999.900000 |  -4,040.000000
-999,999.900000 |  -4,040.000000
```

```
COL A: START UP | COL B:    JAN 83
-99,999,999.9000 |          20.0000
-99,999,999.9000 |       1,900.0000
-99,999,999.9000 |         840.0000
-99,999,999.9000 |         180.0000
-99,999,999.9000 |       4,000.0000
-99,999,999.9000 |          70.0000
-99,999,999.9000 |         850.0000
-99,999,999.9000 |       5,940.0000
-99,999,999.9000 |      -4,040.0000
-99,999,999.9000 |      -4,040.0000
```

Is that all there is to it? Not quite. Remember that *SUPER SST* uses (internally) a value of -999,999,999.90 whenever a output cell is in an unused (or invalid) state. If you reduce the magnitude of the display format, then a number of this magnitude would cause an error message when the new USING statement was encountered. To guard against this possibility, it is necessary to test for absolute values that exceed the new display format. This is relatively easy to do. But, it needs to be done at two places in the program.

One place is immediately after line 4125 (and before line 4130 is executed). Here is how this may be done:

```
4127 V=999999.9:IF ABS(A)>V LET A=-V
```

Remember, this is specifically sized for the example case being illustrated here (with six digits after the decimal place).

The same kind of test needs to be made immediately after the rounding statement in line 6170 *but before the LPRINT statement that ends that line!* Thus, to accomplish this, the LPRINT statement should be moved to a new line (6175, for instance). The above test can then be tacked to the end of line 6170.

To summarize, to change *SUPER SST* so that it displays to six decimal places instead of two, modify the program by altering/adding the six lines shown in the accompanying listing. If you want it to display to four decimal places, use the alternate example listing. To make it display to your tastes, use the information in this discussion and the examples to tailor it to your satisfaction.

IN CASE OF DIFFICULTY WHEN USING PARENTHESES

All registered *SUPER SST* owners who might be affected were notified by mail, but we will repeat the information here in case anyone did not receive the message.

It seems that an incorrect master tape was used to produce tape batches whose serial numbers ended in the following groups:

613041 – 613190
613451 – 613500
623601 – 623605

Tapes within these groups had an incorrect byte in their machine language code. This resulted in the program not being able to properly evaluate expressions that used parentheses. The problem resulted because address &3AC8 contained the value &11 instead of &0B.

All registered owners who had tapes with serial numbers in the ranges indicated were sent information on correcting the problem by performing the following operation:

With SSTMLF loaded into memory, execute:

```
POKE &3AC8,&0B
```

Save the revised SSTMLF portion on tape by executing:

```
CSAVE M "SSTMLF";&38C5,&3FFF
```

Use the updated SSTMLF program in the future.

Owners who did not want to make this alteration themselves were invited to return their tapes for updating.

AVOIDING LOCK-UP OF YOUR POCKET COMPUTER

Apparently some people have not taken our admonitions on page 5 of the *SUPER SST Instruction Manual* seriously. It is there that we warn users not to use the PC for other purposes if they want to keep using a spreadsheet that is stored in memory. Failure to heed this advice can result in the PC "locking-up" so that it is necessary to push the reset button, thereby losing everything in memory. Trying to use *SUPER SST* after you have been using the PC for other purposes *without having issued a RUN directive, thereby re-initializing all variables*, can indeed cause a system lock-up. Please, heed the advice we gave in the operating instructions. It was meant to save you the trouble of having to reload the program and/or data!

FUNDAMENTAL PROFIT/LOSS PROJECTIONS

The accompanying examples illustrate the use of *SUPER SST* in making "what if" projections concerning the operations of a typical small business operation. The first example presents the types of formulas one might want to set up in order to examine a business situation. The second example illustrates the output one would receive for the specific values used in the first illustration.

The example analyzes operations over the course of a full year on a monthly basis. It is assumed (Row 0) that the firm sells some kind of device (widgets?) and plans on selling a certain (increasing) amount each month. Row 1 is used to determine gross sales by multiplying the number of items sold each month by the gross price (assumed to be $95.00 here).

The next five rows are used to calculate expenses. Note (Row 2) that initially the production costs are set at $42.00 per unit. However, later in the year as volume increases, this amount is reduced by a discounting factor that the business expects to receive from its vendors. Similarly, production costs (labor) is expected to decline during the latter part of the year as employees are trained and become more proficient at producing widgets. In other areas, such as advertising and shipping, management plans to increase spending at a steady rate. The advertising budget is to expand by 10% per month. Shipping costs are expected to start increasing later in the year. Finally, (Row 6), an inflation factor is figured into the costs of overhead!

Row 7 calculates total expenses each month by simply summing rows 2 through 6 using the special *SUPER SST* summation operator (invoked by the = sign).

Obtaining a monthly profit or loss projection is then simply a matter of subtracting total expenses (Row 7) from sales revenue (Row 1). A cumulative profit/loss figure is obtained (after the first month) by adding the previous cumulative p/l figure to the current month's results!

The last column in the spreadsheet is used to obtain a total for each classification by summing across the rows, except for the cumulative profit/loss row. Here it is more appropriate to effectively cross-check the layout by subtracting total expenses from total sales. It should agree with the cumulative p/l figure obtained in cell M9.

Example *Data Matrix for a Typical Small Business Profit/Loss Projection*

* 83 PROJECTIONS	COL A: START UP	COL B: JAN 83	COL C: FEB 83
ROW 0: UNIT SLS	0	20	50
ROW 1: DOLLARS	0	95*B0	95*C0
ROW 2: MATERIAL	0	42*B0	42*C0
ROW 3: PRODUCTN	0	B0*9.00	C0*9.00
ROW 4: ADVRTSNG	0	4000	B4*1.1
ROW 5: SHIPPING	0	B0*3.5	C0*3.5
ROW 6: OVERHEAD	0	850	B6*1.005
ROW 7: TTL EXPS	0	B2=B6	C2=C6
ROW 8: MNTH P/L	0	B1-B7	C1-C7
ROW 9: CUMU P/L	0	B8	B9+C8

COL D: MAR 83	COL E: APR 83	COL F: MAY 83	COL G: JUN 83
100	120	144	173
95*D0	95*E0	95*F0	95*G0
42*D0	42*E0	42*F0	42*G0
D0*9.00	E0*9.00*.95	F0*9.00*.95	G0*9.00*.95
C4*1.1	D4*1.1	E4*1.1	F4*1.1
D0*3.5	E0*3.5	F0*3.5	G0*3.5
C6*1.005	D6*1.005	E6*1.005	F6*1.005
D2=D6	E2=E6	F2=F6	G2=G6
D1-D7	E1-E7	F1-F7	G1-G7
C9+D8	D9+E8	E9+F8	F9+G8

COL H: JUL 83	COL I: AUG 83	COL J: SEP 83	COL K: OCT 83
207	249	299	358
95*H0	95*I0	95*J0	95*K0
(42*.95)*H0	(42*.9)*I0	(42*.85)*J0	(42*.8)*K0
H0*9.00*.9	I0*9.00*.9	J0*9.00*.9	K0*9.00*.85
G4*1.1	H4*1.1	I4*1.1	J4*1.1
(H0*3.5)*1.05	(I0*3.5)*1.05	(J0*3.5)*1.05	(K0*3.5)*1.05
G6*1.005	H6*1.005	I6*1.005	J6*1.005
H2=H6	I2=I6	J2=J6	K2=K6
H1-H7	I1-I7	J1-J7	K1-K7
G9+H8	H9+I8	I9+J8	J9+K8

COL L: NOV 83	COL M: DEC 83	COL N: YTD TTLS
430	516	B0=M0
95*L0	95*M0	B1=M1
(42*.75)*L0	(42*.7)*M0	B2=M2
L0*9.00*.85	M0*9.00*.85	B3=M3
K4*1.1	L4*1.1	B4=M4
(L0*3.5)*1.05	(M0*3.5)*1.05	B5=M5
K6*1.005	L6*1.005	B6=M6
L2=L6	M2=M6	B7=M7
L1-L7	M1-M7	B8=M8
K9+L8	L9+M8	N1-N7

Example *Results Matrix for a Typical Small Business Profit/Loss Projection*

* 83 PROJECTIONS	COL A: START UP	COL B: JAN 83	COL C: FEB 83
ROW 0: UNIT SLS	0.00	20.00	50.00
ROW 1: DOLLARS	0.00	1,900.00	4,750.00
ROW 2: MATERIAL	0.00	840.00	2,100.00
ROW 3: PRODUCTN	0.00	180.00	450.00
ROW 4: ADVRTSNG	0.00	4,000.00	4,400.00
ROW 5: SHIPPING	0.00	70.00	175.00
ROW 6: OVERHEAD	0.00	850.00	854.25
ROW 7: TTL EXPS	0.00	5,940.00	7,979.25
ROW 8: MNTH P/L	0.00	-4,040.00	-3,229.25
ROW 9: CUMU P/L	0.00	-4,040.00	-7,269.25

COL D: MAR 83	COL E: APR 83	COL F: MAY 83	COL G: JUN 83
100.00	120.00	144.00	173.00
9,500.00	11,400.00	13,680.00	16,435.00
4,200.00	5,040.00	6,048.00	7,266.00
900.00	1,026.00	1,231.20	1,479.15
4,840.00	5,324.00	5,856.40	6,442.04
350.00	420.00	504.00	605.50
858.52	862.81	867.13	871.46
11,148.52	12,672.81	14,506.73	16,664.15
-1,648.52	-1,272.81	-826.73	-229.15
-8,917.77	-10,190.59	-11,017.31	-11,246.47

COL H: JUL 83	COL I: AUG 83	COL J: SEP 83	COL K: OCT 83
207.00	249.00	299.00	358.00
19,665.00	23,655.00	28,405.00	34,010.00
8,259.30	9,412.20	10,674.30	12,028.80
1,676.70	2,016.90	2,421.90	2,738.70
7,086.24	7,794.87	8,574.36	9,431.79
760.73	915.08	1,098.83	1,315.65
875.82	880.20	884.60	889.02
18,658.79	21,019.24	23,653.98	26,403.96
1,006.21	2,635.76	4,751.02	7,606.04
-10,240.26	-7,604.50	-2,853.48	4,752.55

COL L: NOV 83	COL M: DEC 83	COL N: YTD TTLS
430.00	516.00	2,666.00
40,850.00	49,020.00	253,270.00
13,545.00	15,170.40	94,584.00
3,289.50	3,947.40	21,357.45
10,374.97	11,412.47	85,537.14
1,580.25	1,896.30	9,691.33
893.47	897.94	10,485.23
29,683.19	33,324.50	221,655.14
11,166.81	15,695.50	31,614.86
15,919.37	31,614.86	31,614.86

P.O. Box 232, Seymour, CT 06483

Information provided in this publication is intended solely for the use of registered *SUPER SST* program owners and is presented without warranty of any kind.